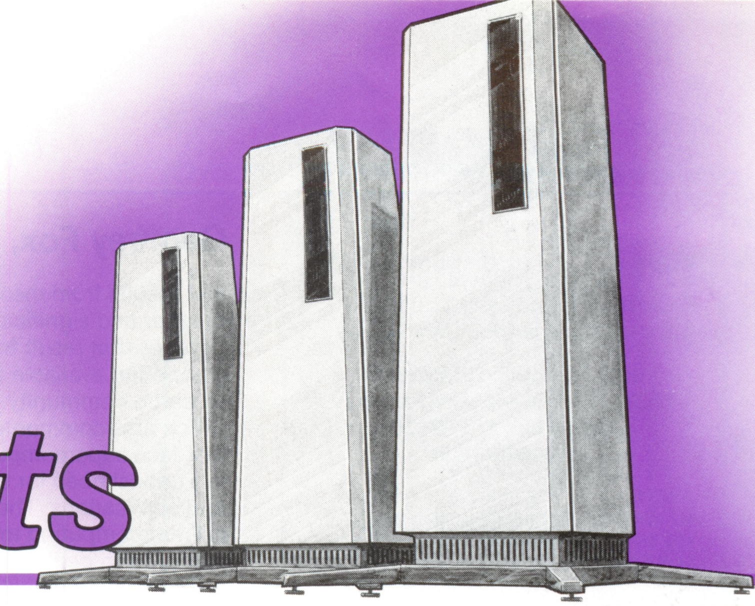
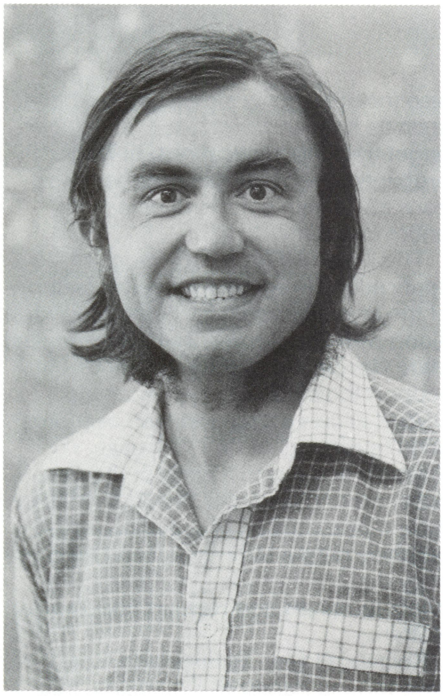


iSCurrents

Published by Intel Scientific Computers

Summer 1987

Computational Science at Caltech: An Interview With Geoffrey Fox



Dr. Geoffrey Fox is Associate Provost for Computing at California Institute of Technology, and Director of the Caltech Concurrent Computation Program (C3P).

"We're at a watershed," said Dr. Geoffrey Fox, Associate Provost for Computing at California Institute of Technology and one of the men who has put hypercubes at the forefront of concurrent computing. "Although much theoretical *computer science* remains to be done, we now have the necessary tools to do *computational science* — that is, to apply real supercomputing power to solve current research problems in science and engineering."

Fox is one of the most active members of the concurrent computing community. His initial interest in the hypercube grew out of the need for more computing power to solve numerical problems in high-energy physics. Fox collaborated with Dr. Charles Seitz and others during development of the original hypercube design at Caltech, with research partially funded by Intel Corporation. Seitz designed and built the first hypercube (the "Cosmic Cube"), and Caltech/Jet Propulsion Lab have constructed two additional generations of the design.

The emphasis on "real computing for real science" sets the tone for work in the Caltech Concurrent Computation Program (C^3P), which Fox directs. "We are refocusing on the uses of supercomputing and on developing software tools to support the full range of scientific research at Caltech. We have now about 40 separate codes running on the hypercube architecture."

C^3P includes the efforts of about 15 people full time, and Fox has asked Paul Messina from Argonne National Labs to become Project Director for the ongoing work of C^3P. Tools are under development to support all pure and applied research at Caltech, including:

- Applied Math and Computer Science: computer graphics, CAD, load balancing algorithms, mathematics and logic
- Biology: modeling of cortex and neural networks
- Chemistry and Chemical Engineering: protein dynamics, chemical reaction dynamics
- Engineering: plasma physics, finite element analysis, condensed matter simulations, vortical flows, image processing
- Geophysics; seismic waves, geodynamics, normal modes of earth
- Physics: fluid jets in astrophysics, high-energy physics, lattice gauge theory, molecular dynamics

Fox's own research interests remain strongly applied: high-energy physics, multiple target tracking, simulation of cortical and neural processes (a cortex simulation is now running at Caltech), load balancing algorithms, and others. He has also found time to coauthor *Solving Problems on Concurrent Processors* with several others at Caltech. The book, scheduled for publication this fall by Prentice-Hall, presents a general approach to concurrent computing and describes the CrOS III — or *crystalline* — implementation of that approach.

This emphasis on using supercomputing tools to solve applied problems in science and engineering is a key feature of C^3P, according to Fox. "The success of this program will not be measured by the number of academic papers published, but by the quality of the science

(continued p.2)

In this Issue of iSC

iSCurrents

Volume 2 Summer 1987

iSCurrents is published quarterly (Spring, Summer, Fall, Winter) by:

Intel Scientific Computers
15201 N.W. Greenbrier Parkway
Beaverton, OR 97006
503-629-7629

Geoffrey Fox, (cont.)

that results from these projects, by the number and significance of scientific discoveries that result from making supercomputing available to the Caltech research community."

Fox also commented on the challenges that face supercomputing today. "There is a critical need for software, both tools and applications. The best thing for concurrent computing today would be for someone to dedicate $15 million a year *just to converting existing codes*. We have 30 years of scientific applications written for sequential machines. It's going to take time and money to bring those into the concurrent arena. It's not very glamorous work, but it is absolutely necessary if concurrent computing is to become commercially viable."

Summer and Fall Training Schedule
Programming Concurrent Computers

July 6-10
August 10-14
September 21-25
October 19-23

For additional information, contact Intel Scientific Computers, Training Center, 15201 N.W. Greenbrier Parkway, Beaverton, OR 97006, 503-629-6279.

Do you know anyone . . .

. . . who might be interested in receiving ***iSCurrents***. If so, please contact:

iSCurrents
Intel Scientific Computers
15201 N.W. Greenbrier Parkway
Beaverton, OR 97006

CrOS III —
A High-Performance Communications System For Hypercubes

Simplicity and speed are the keys to CrOS III, the latest version of the *crystalline* node-to-node communications system for hypercubes. The crystalline system was developed by Dr. Geoffrey Fox and his associates at California Institute of Technology. "Call it stripped down or call it elegant," said Fox. "Either way, CrOS III provides the level of performance and portability that are essential to develop a broad range of applications software for the hypercube architecture."

CrOS III is a set of functions designed to optimize node-to-node message passing in a hypercube architecture. It trades some of the functionality of a full node executive (such as the iPSC system's NX) for increased speed and efficiency. CrOS III is available for iPSC systems and is applicable to a wide range of scientific applications. Support for CrOS III is available from Caltech and from ParaSoft, a newly formed Los Angeles company that provides commercial software for the iPSC system and other hypercube environments.

Adam Kolawa, a recent Ph.D. under Fox and now on the staff at Caltech, is one of the developers of CrOS III. "CrOS III is fully compatible with the iPSC system's NX programming environment," said Kolawa. "Any program that runs under NX can be easily modified to run with CrOS III under NX, with significant speedups. Software conversion is easy. For nearest neighbor messages, just change CALL names for message send and receive routines. For longer distance messages, the Crystal Router utility simplifies conversion."

The crystalline system was initially developed to match the problem topologies that Fox encountered in his research work in high-energy physics. "You gain the most from concurrency when the machine's interconnection topology matches the natural topology of the problem," said Fox. "Many problems in the physical sciences are characterized by 'nearest-neighbor' communications in a domain with regular geometric decomposition. Grid methods for boundary value problems are the classical example. But you see the same structure in image processing and other areas. The name *crystalline* came from this geometric regularity."

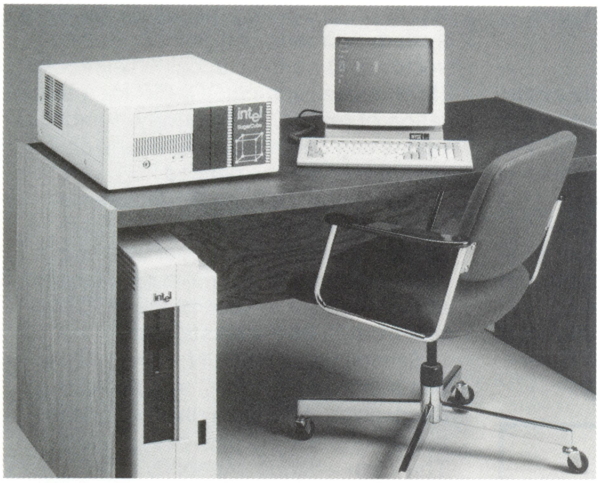
But crystalline has turned out to be much more general than first suspected, said Fox. "Crystalline works well for any problem that is 'loosely synchronous'. That is, there is a compute stage during which nodes compute independently and in parallel, then a communicate stage in which the nodes communicate results of the compute stage among themselves, then another compute stage, and so on. This includes geometrically regular problems such as grid-based numerical methods and image processing. But it also includes many simulation methods such as Kalman filters, used in multiple target tracking applications. In fact, probably 90% of the scientific applications at Caltech could run efficiently under crystalline."

This potential for a wide range of applications is another attractive feature of the crystalline system. Adam Kolawa, along with Jon Flower, John Salmon, and Marc Goroff, are developing additional support tools for CrOS applications. *CUBIX* allows a single version of source code to run on both a sequential machine and a single node of the hypercube, making program development easier and faster. *PLOTIX* provides vector graphics output to both soft and hard copy devices, including standard Tektronix terminals and Hewlett Packard plotters.

Kolawa, Flower, Salmon and Goroff have formed ParaSoft, a company that will provide commercial support for CrOS, CUBIX, and PLOTIX. "We believe that the CrOS approach provides the speed and versatility needed to build a large base of hardware-independent applications codes," said Flower. "We are committed to providing CrOS support for both current and future Intel machines."

Noncommercial versions of CrOS III, CUBIX, and PLOTIX, including iPSC versions, can be obtained from Adam Kolawa, 206-49 California Institute of Technology, Pasadena CA 91125, 818-356-2907.

SugarCube™ System Sweetens Entry Into Concurrent Computing



The SugarCube™ System offers a complete concurrent workstation for as little as $44,950. SugarCube™ systems serve as excellent entry-level machines for educational needs, as well as development workstations for large-scale concurrent computation and OEM applications.

"With the rapidly expanding interest in concurrent computing, there's a critical need for an exploration tool priced under $50,000," said Paul Wiley, SugarCube™ system product manager at iSC. "That's why we developed the SugarCube™ system." The SugarCube system offers a complete concurrent workstation for $44,950, with a vector workstation for under $70,000. And applications developed on the SugarCube system are binary-compatible with larger iPSC systems, making it an excellent entry vehicle into concurrent computing, as well as an ideal development workstation.

The SugarCube system comes in four configurations: basic eight-node system, extended memory system for AI research, vector system, and hybrid (vector and extended memory) system. The AI and hybrid systems include Concurrent Common LISP from Gold Hill Computers at no additional cost.

The SugarCube system is an extension of the iPSC family of 80286-based concurrent computers, but repackaged for size and cost-effectiveness. Each SugarCube system includes a multi-node hypercube, a Cube Manager, and a full range of development tools. Standard tools include the "NX" node executive, a concurrent debugger, communication routine library, FORTRAN, C, and the iPSC system SIMULATOR for off-line program development. In addition, CCLISP is standard on the AI and hybrid systems, and the VAST-2 FORTRAN Vectorizer from Pacific Sierra Research is available at additional cost for the vector and hybrid systems. (See related article on development tools.) Standard service, support, and training are also included with SugarCube.

Wiley sees several uses for SugarCube systems. "First, the SugarCube™ system is an excellent place to begin exploring concurrent computing for a minimal initial investment. And with *complete upward compatibility* with the larger iPSC systems, that investment is well protected." As the customer expands to larger iPSC systems, the SugarCube system continues to serve as a program development workstation. Completed programs can then be run with full data or a full complement of processing nodes on the larger iPSC system.

"In addition, more educational institutions can now afford concurrent computing," said Wiley. A typical university scenario includes students doing routine coursework with iPSC SIMULATOR packages on UNIX- or XENIX- based PCs, VAXs, or workstations. The SugarCube system is then used for projects requiring speed or actual hardware-based timing measurements.

The SugarCube system also makes an ideal development station for current large-system iPSC users. SugarCube systems can attach to the same TCP/IP Ethernet, and source and binary programs can be exchanged with a larger, central iPSC system, providing a cost-effective way to reduce the user load on current iPSC systems.

Finally, the SugarCube system's price/performance makes it an excellent choice for OEM use in production computing applications. "We expect the SugarCube™ system to stimulate interest in concurrent computing among OEMs, for both individual and shared systems," said Wiley.

SugarCube configurations are:

- iSGR/d3—basic system of eight standard nodes, each with 512K local memory. Designed for general purpose concurrent computing applications and research.
- iSGR-MX/d2—four memory nodes with 4.5 MBytes of memory each. Designed for the large programs and data structures required in symbolic and AI applications. Includes Gold Hill's CCLISP.
- iSGR-VX/d2—a vector system with four vector nodes. Designed for large-scale compute-intensive applications, with a peak performance of 26 MFLOPS.
- iSGR-HX/d2—a hybrid system with two VX vector nodes and two MX memory nodes. Designed for applications that require both numeric and symbolic processing. Includes Gold Hill's CCLISP.

For details on SugarCube systems, contact Intel Scientific Computers, 15201 N.W. Greenbrier Parkway, Beaverton, OR 97006, 503-629-7629.

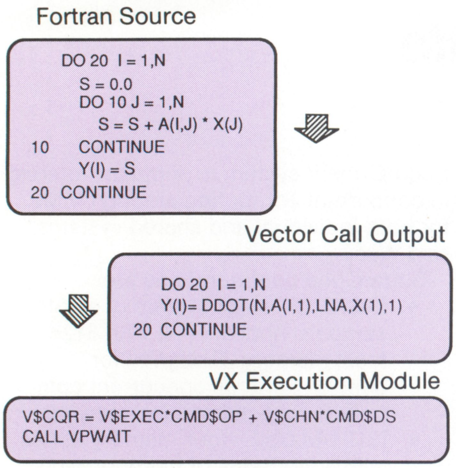
iPSC Software Development Tools

The iPSC software development environment offers a well-stocked tool kit to aid software developers. Some of the most important tools are described here.

The VAST-2 Vectorizer automatically vectorizes sections of FORTRAN code for execution on a vector node (see illustration). This means easier conversion of existing code for vector machines, and easier, more efficient programming. In addition, system vector performance can be optimized with shorter vectors than before. VAST-2 is produced by Pacific Sierra Research, a well-known developer of vectorizing software for supercomputing environments.

The iPSC *SIMULATOR* software package simulates the iPSC concurrent environment on machines running XENIX or Berkeley UNIX. Priced under $500, the

(continued p.4)



VAST-II VECTORIZER

(continued from p.3)

SIMULATOR package offers a method of exploring concurrent computing on existing sequential systems, with a very small initial investment. SIMULATOR is included with all iPSC and SugarCube systems.

The iPSC system's concurrent debugger, is the first commercially available symbolic debugger for large-scale distributed-memory concurrent machines. In addition to conventional process-based symbolic debugging capabilities (monitor and control any process on any node), the debugger provides examination, tracing, and control of message traffic between nodes. A FORTRAN version is included in the current operating system (Release 3.1). A C version will be included in the next system release (4.0), later this year.

System Release 3.1 also supports *hybrid cube* capability. This allows a single hypercube to contain both vector and symbolic nodes, giving access to two different programming models in the same system. Numeric computations using vector (VX) nodes can call on symbolic (MX) nodes running CCLISP for interpretation of results. Alternatively, symbolic nodes can control computations on numeric nodes, as needed.

In addition, Release 3.1 includes a prototype version of a *remotely connected host* facility for user evaluation and comment. The remote host facility allows the user to control the iPSC system from SUN-3 workstations.

Several users on SUN workstations can subdivide the iPSC system among themselves, with each user assigned a subset of the iPSC processing nodes. This *cube sharing* capability allows users to maximize utilization of cube resources, apportioning the cube out to several smaller tasks or dedicating it to a single user for larger tasks.

For more information about any of these tools, contact Intel Scientific Computers, 15201 N.W. Greenbrier Parkway, Beaverton, OR 97006, 503-629-7629.

SugarCube™ System Is Ideal Educational Package

With the announcement of the SugarCube system, Intel Scientific Computers now offers the ideal entry-level package for educational concurrent computing needs:

- The SugarCube system serves as a low-cost concurrent computer, well suited to classroom computing needs.
- The iPSC SIMULATOR software package, included with the SugarCube system, allows students to prototype codes on a variety of familiar UNIX and XENIX workstations. These codes can then be run on the SugarCube system for true concurrent processing.
- Each SugarCube system includes one training credit for iSC's *Programming Concurrent Computers* workshop, a five-day training class that provides lectures and hands-on lab sessions to build concurrent computing expertise.

In addition the CrOS III Communications system from Caltech runs on all SugarCube systems, and is documented in *Solving Problems on Concurrent Processors* by Dr. Geoffrey Fox, et al.

In Future Issues . . .

. . . of ***iSCurrents*** we'll feature iPSC applications in image processing, modeling and simulation, and CAD, plus a special feature on AI and symbolic computing, and other developments in the field of concurrent computing.

Intel Scientific Computers
15201 N.W. Greenbrier Parkway
Beaverton, OR 97006

Address Correction Requested